Radiation Fault Modeling and Fault Rate Estimation for a COTS Based Space-borne Supercomputer

Arbi V. Karapetian Arbi.V.Karapetian@jpl.nasa.gov, Raphael R. Some Raphael.R.Some@jpl.nasa.gov, John J. Beahan John.J.Beahan@jpl.nasa.gov
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109
(818) 393-4770

Abstract—Development of the Remote Exploration and Experimentation (REE) Commercial Off The Shelf (COTS) based space-borne supercomputer requires a detailed model of Single Event Upset (SEU) induced faults and fault-effects. Extensive ground based radiation testing has been performed on several generations of the Power PC processor family and related components. A set of relevant environments for NASA missions have been analyzed and detailed. Combining radiation test data, environmental data and architectural analysis, we have developed a radiation fault model for the REE system. The fault model is hierarchically organized and includes scaling factors and optional parameters for fault prediction in future technologies and alternative architectures. It has been implemented in a generic tool, which allows for ease of input and straight forward porting. The model currently includes the Power PC750 (G3), PCI bridge chips, L2 Cache SRAM, Main Memory DRAM, and the Myrinet packet switched network. In this paper, we present the REE Radiation Fault Model and accompanying tool set. We explain its derivation, its structure and use, and the work being done to validate it.

TABLE OF CONTENTS

1. INTRODUCTION

Future NASA missions will require onboard supercomputing capabilities for both science data processing and for autonomous robotic operations. Constraints in available bandwidth preclude transmission of high volumes of raw science data to earth, currently limiting some missions to as little as 5-10% of their potential science return. Furthermore, round trip communication delays and the high cost of manual vehicle control severely impact mission operations of robotic vehicles by limiting both the mission's planned objectives and the potential for "opportunistic science". Overcoming these limitations through the provision of state of the art supercomputing onboard NASA spacecraft would greatly enhance the science return from NASA space exploration missions.

Leveraging the substantial commercial investment in COTS supercomputing technology could provide 10-100X improvements in computing capability over the radiation-hardened computers currently used in space-borne systems. However, while state of the art COTS components are in general sufficiently hard to Total Ionizing Dose (TID) to survive a 5-10 year mission in most of the environments of interest to NASA missions, Galactic Cosmic Rays (GCRs) and energetic protons (mostly solar) cause SEUs, in the form of soft errors (random bit flips) in COTS components.

The REE project seeks to develop a supercomputing system based on inexpensive, state of the art, COTS components and Software Implemented Fault Tolerance (SIFT) techniques, which will allow the system to tolerate SEUs without a significant penalty in power, performance, mass or volume. The system currently being developed is a Power-PC based embedded cluster computer. The Operating System (OS) is a commercially available file system based (i.e. Linux/Unix-type) OS, and the high speed interconnect fabric is Myrinet. The architecture has been implemented in a variety of clusters at JPL to ensure portability and ease of upgrade. REE testbed clusters include: PPC603, PPC604 and Pentium III processor based nodes; 100Mbit Ethernet and Myrinet based network interconnects; Linux and Lynx OS's; and several versions of the PMI message passing protocol. In addition, multiple onboard science applications supporting a variety of missions have been developed and ported to these systems.

Figure 1 shows the top-level view of a typical instantiation of an REE system intended for deep space missions. It is similar in concept to a commercial cluster processor comprising a set of processing nodes interconnected by a high-speed switched packet network (Myrinet). Mass memory nodes replace commercial disks, providing a shared file system. The mass memory nodes also implement the operational interface to the rest of the spacecraft via the Spacecraft High Speed Data Bus. In addition a "back door bus" provides a path for the (radiation hardened) spacecraft control computer to reset or interrogate the cluster nodes as well as for programmers and system engineers to test and debug the system during development or during mission operation.

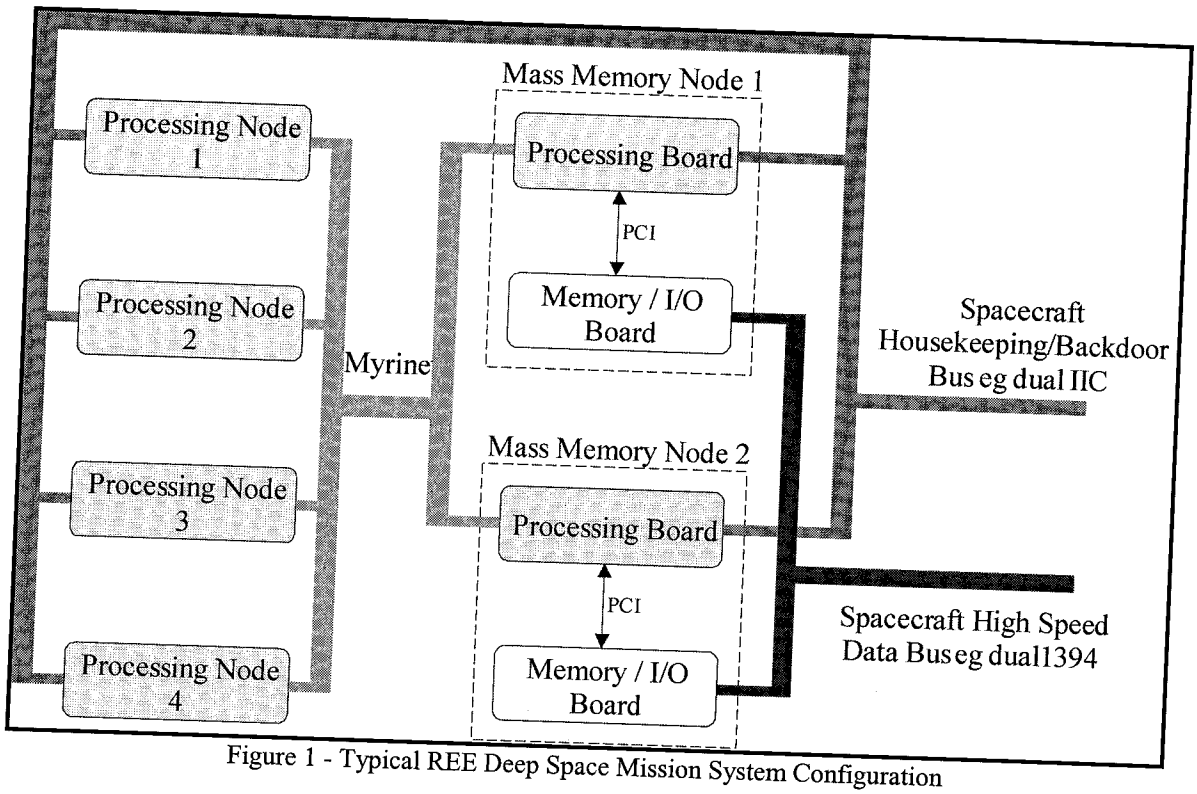

Figure 1 - Typical REE Deep Space Mission System Configuration

Development of the REE architecture and SIFT techniques requires a detailed model of SEU induced faults and fault-effects. Extensive ground based radiation testing has been performed on several generations of the PPC family and related COTS components. A set of relevant environments for NASA missions have been analyzed and detailed. Combining radiation test data, environmental data and architectural analysis, we have developed a fault model of a REE-type system, initially described in [1] The fault model is hierarchically organized and includes scaling factors and optional parameters for fault prediction in future technologies and alternative architectures. The fault model has been implemented in a generic tool, which allows for ease of input and straightforward porting. It currently includes the PPC750 (G3), PCI bridge chips, L2 Cache SRAMS, Main Memory DRAM, and the Myrinet packet switched network.

The remainder of the paper is organized as follows. Section 2 of this paper explains the overall methodology and tool set developed on the REE project to support system reliability and fault behavior analysis. Section 3 describes the radiation fault model at the system, node, and component levels. Section 4 describes the way in which the fault model is verified, including the verification of radiation test data and verifying the translation of thresholds to environments. Section 5 provides fault rates for various space environments for the REE First Generation Testbed (FGT), as computed by the radiation fault model. Section 6 provides conclusions and suggestions for future work.

2. METHODOLOGY OVERVIEW

The REE project requires a means for trading off performance and power utilization versus reliability and availability. The method must be generally applicable to alternative architectures and applications and, once developed, relatively straightforward to implement. Unlike traditional fault tolerant systems, a degree of unreliability or unavailability is acceptable, i.e., .95 or .99 rather than .99999 is an acceptable reliability figure for REE. On the other hand, it is imperative that the system fault behavior and reliability be accurately predictable. The mission system engineer must be able to 'dial in' a desired level of reliability and fault behavior based on mission phase and criticality. Thus, a methodology is required which will allow characterization and modeling of probabilistic system behavior, reliability and availability under varying applications, environments, loads, and operational scenarios.

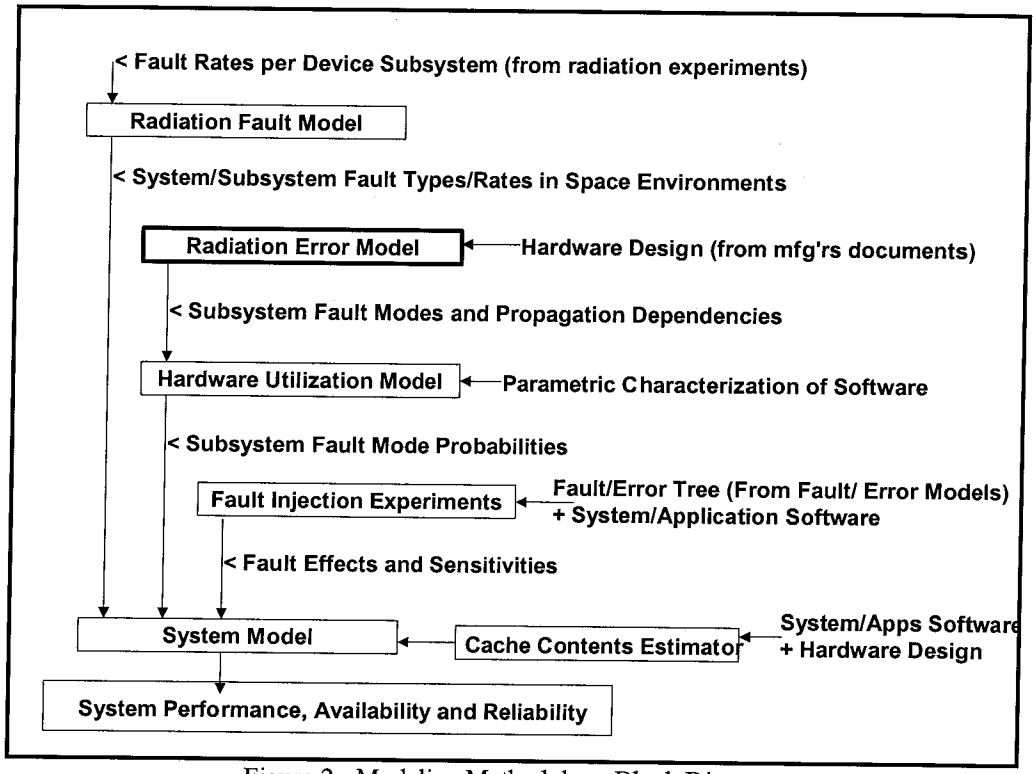

Figure 2 - Modeling Methodology Block Diagram

Figure 2 shows the methodology and tool set developed for the REE project.

Radiation effects experiments are performed on the hardware components to determine subsystem level radiation sensitivities. Results for a processor, for example, include fault rates for the L1 data cache, the L1 instruction cache, the General Purpose Registers (GPRs), the Floating Point Registers (FPRs), the memory management unit (MMU), etc.

The results of the radiation experiments are used to develop a radiation fault model. This model is used to predict the fault rates that will occur in a given radiation environment (e.g., Low Earth Orbit, Geo-synchronous Orbit, Deep space, Solar Flare, etc.). The model provides the number of faults per unit time per subsystem.

Using information about the hardware architecture, the Error Model predicts the types of errors that can arise as a result of an SEU occurring in a given subsystem. Essentially, the process of generating the error model is one of listing all possible faults and then, by analysis, propagating each fault through the hardware to the first point at which it impacts software or system operation. The emphasis of this effort is on subsystems into which faults cannot be directly injected with Software Implemented Fault Injection (SWIFI). Thus, it is not necessary to trace every possible error resulting from a general -purpose register bit flip. It is however, necessary to list all the possible outcomes of SEUs in MMU and cache address translation registers, cache tag rams, etc.

The Hardware Utilization Model provides a means for determining the software (hardware utilization) dependent probabilistic fault propagation statistics and the method by which SWIFI fault injection techniques can be used to emulate the effects of the underlying fault.

The central component of this methodology is the construction and execution of fault injection campaigns. Fault injection campaigns are designed to provide fault/error sensitivities of the system components. The campaigns are conducted on the operational system, after which the results are analyzed to determine the effects of the faults (e.g., system crash/hang, incorrect result, no apparent effect) and their associated probabilities.

The Cache Contents Estimator (CCE) is used to deal with the inability of SWIFI techniques to inject bit flip faults into the processor's cache memories. Faults are injected into an application's instruction, data, heap, and stack segments in main memory to determine the fault behavior statistics of each type of error. The CCE predicts how much of each of these segments will be in the cache at any given time. The final error rate for each of these segments in cache is proportional to its size.

Finally, the system reliability and performance model is constructed using knowledge of the system architecture, predictions from the fault model, the results of the fault injection experiments and the CCE results. The model predicts the system's reliability and performance in a given radiation environment. It can be used during system

development to identify appropriate system architectures and fault tolerance strategies. During fielded operation, the model can be used to predict the system's behavior in changing circumstances and modify it as appropriate (e.g., increase check-pointing frequency, uplink fault-tolerant linear algebra libraries). Once the basic system model has been created and validated, it is relatively straightforward to input application- software-specific fault behavior statistics, input the mission environmental parameters, and predict system fault behavior and reliability for a range of fault tolerance techniques, thus it also provides an early testbed for supercomputer based mission development.

3. RADIATION FAULT MODEL

The REE radiation fault model provides SEU fault rates for various NASA mission environments. The model details faults rates at the system, node, component and subsystem levels. The model can be tailored for different configurations including alternate node configurations, number of nodes per system and interconnect topology. For a specified system configuration and environment, an element's fault rates are calculated by hierarchically summing the fault rates of its constituent lower level components. The following paragraphs detail the structure of the REE system architecture and the resultant radiation fault model.

For purposes of illustration and explanation, this paper will discuss the radiation fault model for the REE FGT, which is explained below.

System Architecture

The Detailed architecture of the REE FGT is shown in Figure 3. The system comprises 20 processing nodes interconnected by a dual redundant Myrinet switched network fabric. As shown in Figure 3, the interconnect topology is a chordal ring structure chosen for its reliability, ease of implementation, and ease of extensibility. AC power control provides the ability to reset and cycle power to each of each of two chassis containing 10 nodes each. The host computer is a Sun Sparc Station which provides the interface to the system's mass memory (hard disks) and external network.

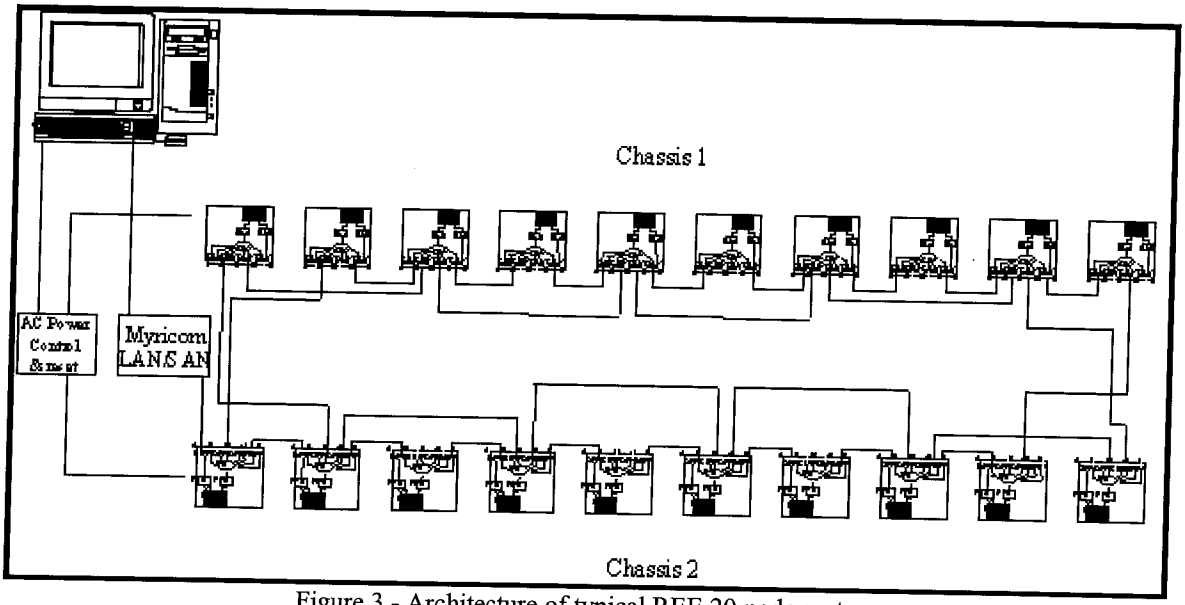

Figure 3 - Architecture of typical REE 20 node system

System Level Fault Model

The fault model, at this level, contains the following elements:

- 20 processing nodes
- System control circuitry such as clock, watchdog, power halt and reset controller

The host computer is not represented in the model, as it would not exist in a fielded configuration. Similarly, the current FGT fault model does not make provision for disk arrays or external networks, as their fault rate would have been unacceptably high and unrealistic.

Figure 4 shows a sample fault estimation page at the system level. The fault rates shown for these system level elements have been calculated from lower levels of the model, i.e., node level, which in turn were calculated from component level SEU rates by using bit-type SEU rates for the specified environment (Deep Space). Count represents the number of elements, such as nodes. Latch Faults/hr are the latch faults rates calculated by summing all functional sub-block components. Gate faults/hr are Gate faults rates calculated by summing all functional sub-block components. Total Faults/hr is a sum of the Latch and Gate fault rates. The Margin column represents a multiplier, and is a safety margin selected based on the uncertainty in the number of latches and gates in the functional block.

System-level design

	Count	Margin	Latch Faults/hr	Gate Faults/hr	Total Faults/hr
System totals:			**139.17**	**0.91**	**140.08**
Number of nodes per system	20		138.91	0.91	139.82
Additional system-level elements		3	0.26	0.00	0.26

Figure 4 - Sample spreadsheet showing System Level Fault Estimation

Node Architecture

A basic diagram of the REE node is depicted in Figure 5. The processors are G3 Power PC750 processors. The Network Interface functions have been delegated to a group of components marked shown in the block marked Network Interface. The L2 Cache has been shown as a separate block, as it resides on external to the CPU chip. The bridge chip is the MPC106, which is the predecessor to the current MPC107 bridge chip. Main Memory is DRAM with Error Detection And Correction (EDAC).

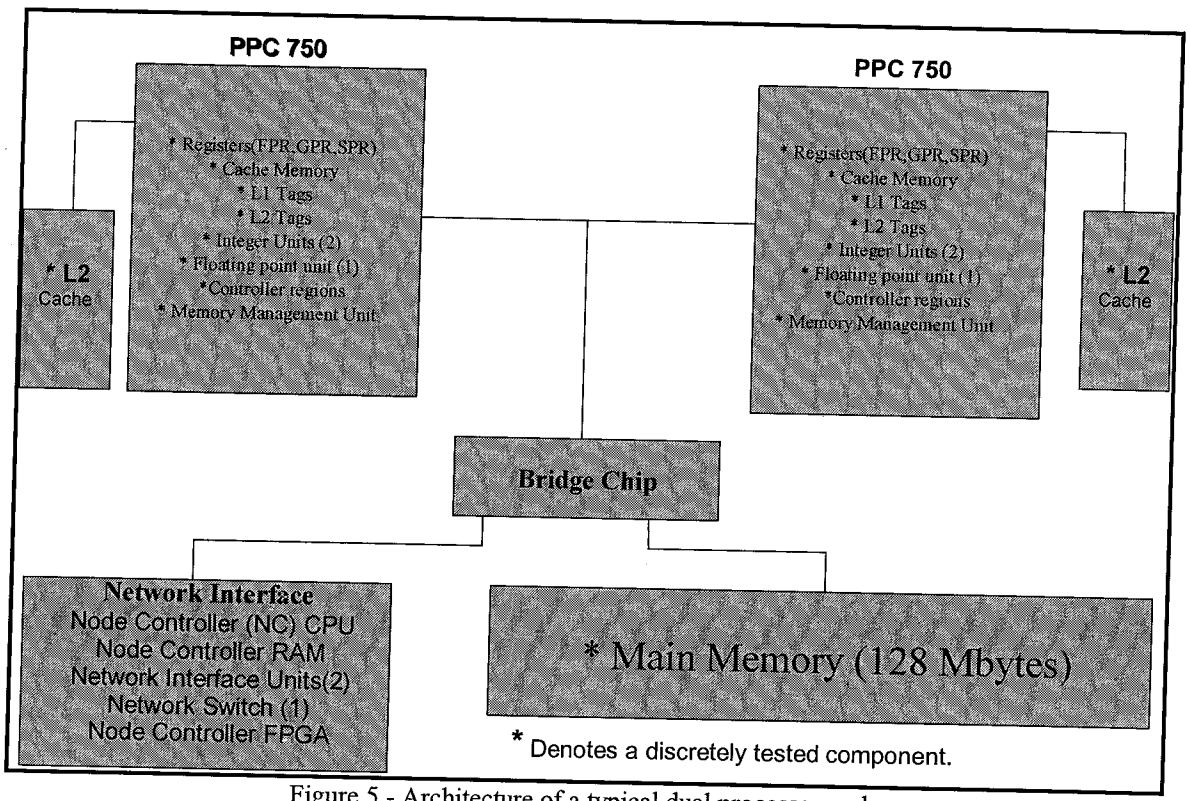

Figure 5 - Architecture of a typical dual processor node

Node Level Fault Model

The next level of the REE fault model is based on the REE Node. Each of the 20 nodes in the REE system is composed of the following components as shown in Figure 5. Note that the node contains the network interface and network switch circuitry: thus, there is no separate network fault model as network faults are reflected in the node fault rate.

- Two G3 Power PC750 (PPC750) RISC microprocessors with off-chip L2 cache
- Motorola MPC106 bridge chip
- 128 Megabytes of Main Node Memory (RAM)
- Network Interface
  - Node controller CPU (Intel StrongArm

SA-110)

- Node controller ram (4 megabytes)
- Myricom network Switch (Myricom xbar8x8)
- Node controller FPGA (Myricom)
- Two network interface units (Myricom LANai 7)
- Misc. elements such as EEPROM, clock, watchdog, Power halt and reset controller

Node-level design

	Count	Margin	Latch Faults/hr	Gate Faults/hr	Total Faults/hr
Totals per node:			**6.95**	**0.05**	**6.99**
Node CPU's per node	2		5.31	0.04	5.36
Node Controller (NC) CPU		1.5	0.71	0.00	0.71
Node Controller RAM		3	0.12	0.00	0.12
Network Interface Units(NIU) per node	2	3	0.35	0.00	0.35
Number of Network Switches per node	1	3	0.16	0.00	0.16
Bus controller (PCI)		3	0.13	0.00	0.13
Misc (watchdog,clock,EEPROM,PHRC)		3	0.02	0.00	0.02
Node Controller FPGA		3	0.14	0.00	0.14

Figure 6 - Sample Spreadsheet Showing Node Level Fault Estimation

Processor Architecture

Due to its complex architecture, the focus of test methodology development, as well as radiation testing in sub node level components has been the PPC750 microprocessor. The PPC750 microprocessor has a complex architecture, with numerous areas not accessible to the user. Figure 7 shows a simplified functional block diagram of the PPC750 as depicted in the PPC750 Users Manual.

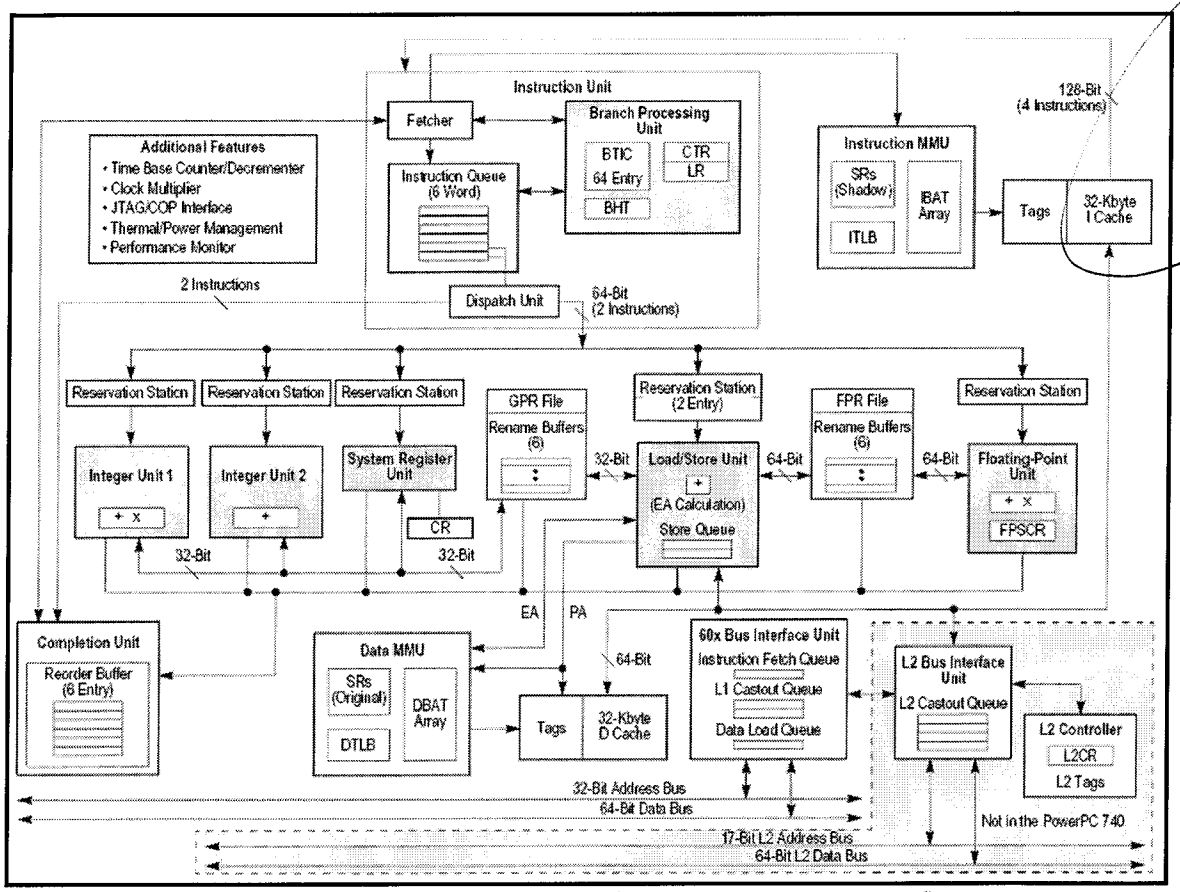

Figure 7 - PPC750 Block Diagram (From PPC750 Users Manual)

Processor Level Fault Model

Due to the unavailability of bit level information in some of the areas of the PPC750, two methods have been utilized for acquiring gate and latch counts for each of the PPC750's functional blocks, which are discussed in section 3.5 of this paper. The second one of these methods is also utilized to estimate gate and latch counts in untested components of the REE system, to produce latch and gate counts, when this information is not available.

In order to make the Fault model scalable for future generations of REE systems and processors, the modeling is done in functional block segments. The current fault model contains the functional blocks for the PPC750 microprocessor. The functional blocks are flowed down to the bit level detail. The functional blocks of the PPC750 are shown in the diagram bellow. They too are listed in the bulletized list below.

- CPU
  - GP registers
  - FP registers
  - SP registers (CR, LR, CTR, XER, FPSCR)
  - Program Counter
  - Control/Status registers
  - Debug registers (decr, watch addr, watch instr)
  - Addressing registers (BIU)
  - Current instruction holding latches
  - Register rename buffers
  - Branch target instruction cache
  - Branch history table
  - MMU/TLBs
  - TLB Tags
  - MMU TLBs
  - BATs
  - Instruction and Data BATs
  - MMU memory segment registers
  - L1 data and instruction cache
  - L1 state bits
  - L1 entries
  - L1 TLBs
  - L1 cast-out queue
  - L2 cache line
  - L2 cache
  - L2 state bits
  - L2 TLBs
  - L2 write-back buffer
  - Hidden latches
    - Instruction sequencer
    - Integer units (2)
    - FPUs
    - Load/store unit
    - Data MMU TLB hidden latches

        - Instruction MMU TLB hidden latches
        - Bus I/F and L2 cache controller hidden latches
        - TAU hidden latch
        - PLL hidden latches
        - JTAG hidden latches
- L1 cache
    - L1 data cache hidden latches
    - L1 instruction cache hidden latches
    - L1 data tag hidden latches
    - L1 instruction tag hidden latches
    - L1 data cache control hidden latches
    - L1 instruction cache control hidden latches
- L2 cache (1 megabyte)
    - Hidden latches (such as JTAG, etc)

Figure 8 depicts a sample of the spreadsheet in the REE Radiation Fault Model for the calculation of the PPC750 rates. Due to the large size of the detailed calculation spreadsheet, the high level spreadsheet is shown.

CPU Level Design (High Level)

				Latch	Gate	Total
		Count	Margin	Faults/hr	Faults/hr	Faults/hr
CPU's per node		2		**5.31**	**0.04**	**5.36**
	Node CPU (w/o L1/L2 cache)		1.5	1.34	0.00	1.34
	CPU L1 cache		1.5	1.08	0.00	1.08
	RAM per node CPU		1.5	0.21	0.02	0.23
	L2 cache per CPU		3	0.03	0.00	0.03

Figure 8 – Sample Spreadsheet showing CPU Level Fault Model (High Level)

Determining component sensitivity

The PPC750 microprocessor is viewed as the set of its building blocks. These building blocks are the basis for the functional block modules in the radiation fault model. SEU characterization of processor bit-types were determined by carrying out radiation experiments in both heavy ion and proton radiation facilities, on isolated functional blocks of the microprocessor. The accessible parts of the microprocessor were tested discretely, using direct testing methods as listed within each component box in Figure 5. These methods can be studied in detail in the following radiation papers [2], [3] and [5]. These SEU characterization experiments revealed two basic bit-types in the PPC750. The two main bit-types are categorized as the register types and the cache memory type bits. The proton and heavy ion SEU rates are then translated into space environment SEU rates by using the AP-8 model for trapped protons and CREME96 for GCRs as environmental models, and using the IRPP method of space rate calculation [4]. The radiation fault model contains the count for register type bits and cache type bits for each functional block, and applies the SEU rates pertaining to these bit-types. As of this time, gate faults have not been observed and thus do not contribute to the overall fault rates of the system. However, the radiation fault model does have provision available in the event of future observations of such effects. Gate fault rates are expected to contribute to the overall fault rate due to shortened clock cycles and smaller feature size. The SEU susceptibility of various functional block of the PPC750 can be found in radiation test reports [2] and [3] The main memory being used for the REE FGT is a standard COTS DRAM, which has been tested extensively in the radiation effects study community. Extensive information about DRAM SEU susceptibility in can be found in [5].

Combining Component Sensitivity with Environment Data

In order to calculate SEU rates in the PPC750, it is necessary to have an accurate count of the number of gates and latches in each internal functional block. Two methods were used to estimate (on the first order) the number of bits in each functional block. The second method is also applied to the rest of the components of the REE Node, (such as the Network Interface chips, the bridge chip, etc.) to estimate their latch and gate counts, when such information is not available. Due to user accessibility limitations to inner

regions of the PPC750, only a subset of the registers was tested discretely. A first order estimation was made of the composition of the remaining functional blocks using information from the PPC750 Users Manual. The following is a list of components, which were tested discretely:

- PPC750 Floating Point Registers (FPR)
- PPC750 General Purpose Registers (GPR)
- PPC750 Special Purpose Registers (SPR)
- PPC750 L1 Tags & Flags
- PPC750 L2 Tags & Flags
- PPC750 Integer Units (IU)
- PPC750 Floating Point Unit (FPU)
- PPC750 Memory Management unit (MMU)
- PPC750 L2 Cache
- Main Memory DRAM

Components yet to be tested are listed bellow:

- MPC106 Bridge Chip
- Node controller CPU (Intel StrongArm SA-110)
- Node controller ram (4 megabytes)
- Myricom network Switch (Myricom xbar8x8)
- Node controller FPGA (Myricom)
- Two network interface units (Myricom LANai 7)
- Inaccessible regions of PPC750

First method:

The PPC750 users manual provides a fair amount of descriptive detail about the components of the processor. This information is used as described below to estimate the number of bits per functional block.

- The count for registers in each functional block is extracted (i.e. GP registers, FP registers etc.).
- This number is then multiplied by the corresponding width (in bits) of these registers in order to obtain a total number of bits for that functional block.

Second Method:

Estimate the number of latches in each functional area of the microprocessor. The number of latches in a region corresponds to the number of bits in that region.

- There are 6.35×10^6 transistors in the Power PC 750.
- Estimate 4 transistors/Gate = 1.6×10^6 Gates. This value is obtained from knowledge of current microprocessor gate construction techniques.
- The percent of total area for each functional block is calculated by measuring the corresponding die overlay region, as shown in Figure 9 and dividing it by the total area. (i.e. Integer Units, Floating Point Unit, etc.)The percent of total area is then multiplied by the total gate count (1.6×10^6) to obtain the number of gates for this functional block.

An average gate-to-latch ratio of 30 gates for each latch is used to calculate the number of latches in that region (first order approximation). Again, the value of the gate to latch ratio is obtained from knowledge of current microprocessor gate and latch construction techniques.

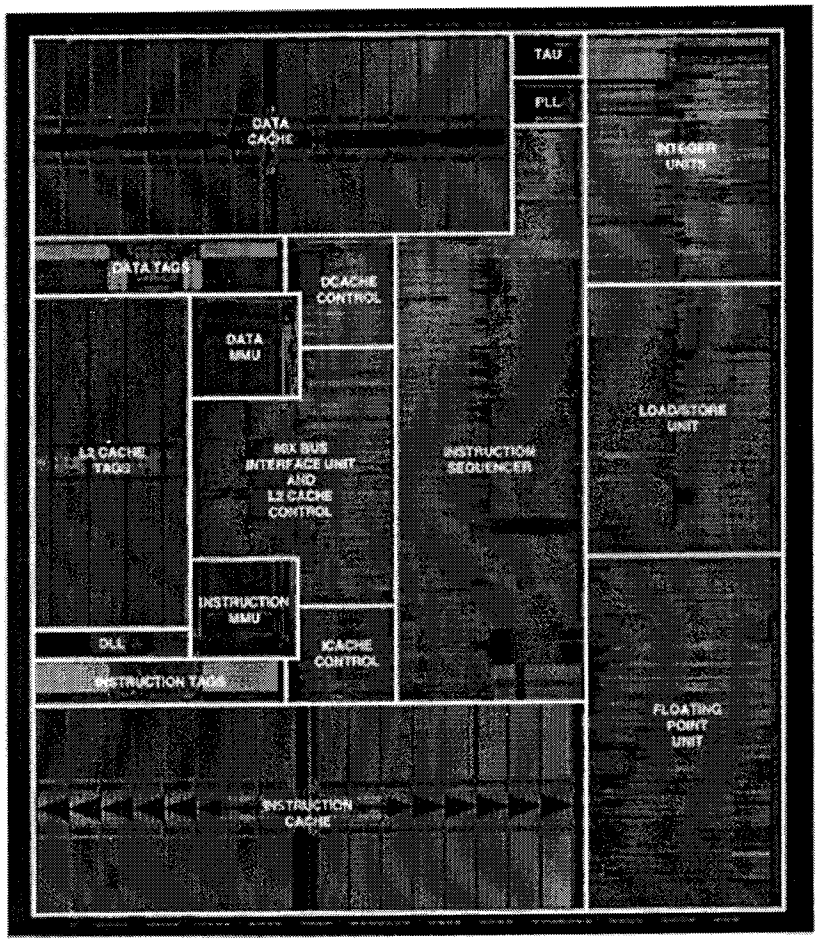

Figure 9- PPC Functional Block Die Layout (From PPC750 User Documentation)

Once the counts for the latches and gates have been determined for a particular functional block, the bit-type of the latches are then determined by the function of the latch to be either cache type bits or register type bits. In the cases where there was uncertainty in determining the bit-type, the bit-type more susceptible to SEUs (Register type) was selected. The latch count is then multiplied by the per-bit SEU rate for the specific bit-type, in each space environment. The per-bit SEU rates for both register and cache bit-types are provided as inputs to the radiation fault model, and are in the format of the chart shown in Figure 10.

SEU Rate for Data Cache *(Per Bit)*				
Orbit or Location	**Environmental Components**	**Sheilding**	**Peak Rate** *(per bit)*	**Average Rate** *(per bit)*
Interplanetary Space	solar min. GCR	- - -	- - -	7.00E-07
	solar max. GCR	- - -	- - -	1.90E-07
	DCF (protons+ions)	60 mil Aluminum	9.20E-03	1.90E-03
		100 mil Aluminum	1.60E-03	3.80E-04
		250 mil Aluminum	7.80E-04	1.80E-04
600km-98°	solar min. GCR	- - -	7.00E-07	1.90E-07
	solar max. GCR	- - -	1.90E-07	6.40E-08
	trapped protons	- - -	1.10E-05	1.80E-07
	DCF (protons+ions)	60 mil Aluminum	9.20E-03	4.80E-04
		100 mil Aluminum	1.60E-03	8.00E-05
		250 mil Aluminum	7.80E-04	3.50E-05
600km-28°	GCR	- - -	- - -	1.70E-08
	trapped protons	- - -	1.10E-05	3.70E-07
Surface of Mars	GCR	- - -	- - -	5.00E-08

Figure 10- Sample Input Chart to the Fault Model for Cache Type Registers

The JPL radiation-test group developed the testing approach for the current SEU rates used in the radiation fault model. The methodology and results are detailed in [2] and [3]. Due to its complex architecture, a good starting point for determining SEU susceptibility in the PPC750 is its register sets. The register set of the PPC 750 is shown in Figure 11.

A sample calculation of the GPR set, from per bit SEU rates, to their contribution to the total fault rate of the PPC750 would be useful. The per-bit SEU rate for the GPRs is derived from experimental data. The registers are loaded with predetermined values (1s or 0s or a mixed pattern), and exposed to radiation, with near-zero processor duty cycle. The per-bit cross-section is then calculated, by taking into account the fluence of the beam particles, particle energies, as well as the number of bits being irradiated. This data is then combined with radiation environment data, to produce per-bit SEU rates for the GPRs. The per-bit SEU rate is used as an input to the radiation fault model in the format shown in Figure 7. The number of GPRs (among other register type bits with similar SEU rates) is then calculated per functional block, and the per-bit SEU rate is multiplied by the number of bits to get a total fault rate for the register type latches in that functional block. The resulting fault rate is then the register bit-type contribution to the overall PPC750 fault rate.

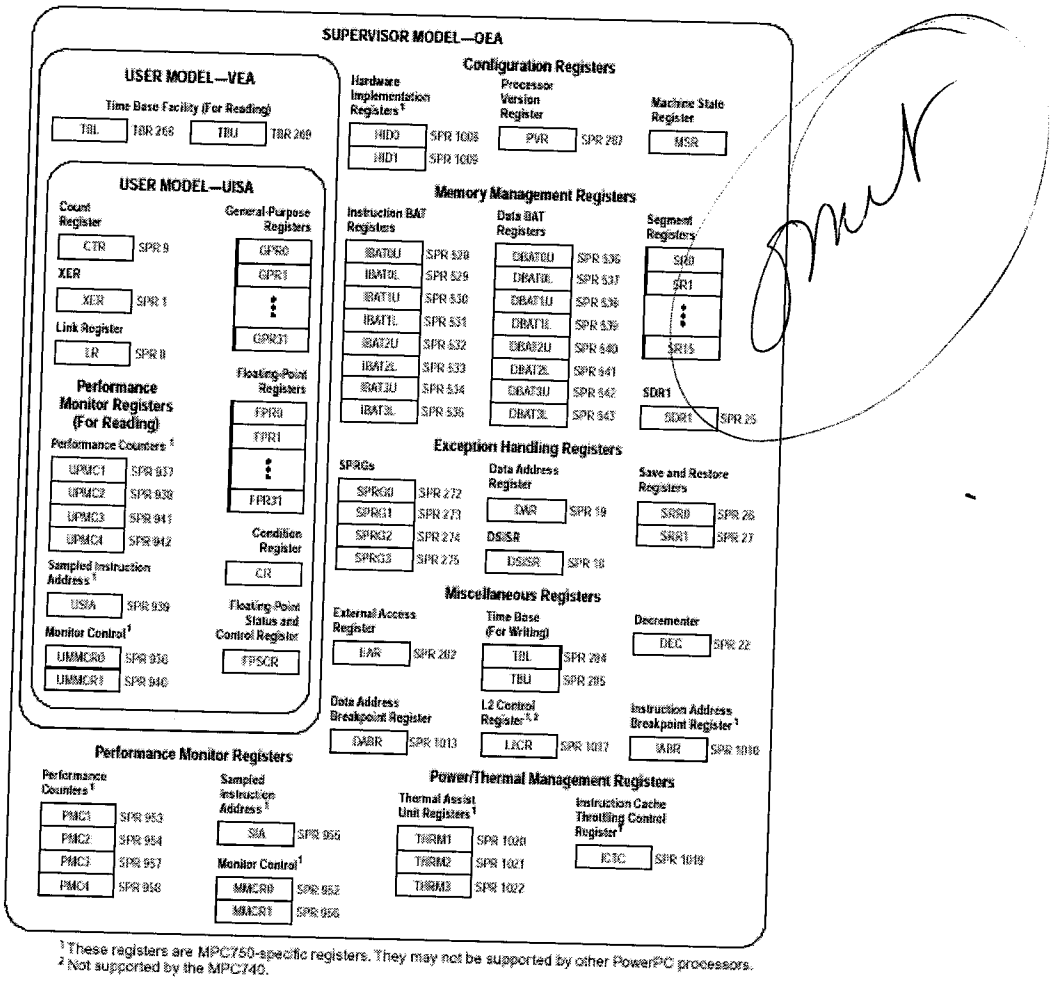

Figure 11- PPC750 Register Set (From PPC750 User Manual)

Assumptions

- Gate Fault Rates have not been observed in laboratory experiments. The fault model currently uses a factor of (0.00001*latch fault rates), as a placeholder. In the event laboratory experiments reveal a different number, or verify a zero fault rate, this number can be changed to update the fault rates.
- The fault rates provided by the radiation fault model are hardware level fault rates only. They require further analysis to determine propagation of errors in software. It is also possible to observe new categories of errors, which are not visible on the hardware fault level.

Currently, there is no empirical data on the SEU rates of the MPC106 bridge chip being utilized in the FGT. An estimate of the latch and gate count is made, by knowing the dye area, the feature size, and the function of the bridge chip. This methodology is also used as a starting point for the Network chipset, in order to estimate their SEU rates.

4. MODEL VERIFICATION

Due to the number of assumptions and estimations made in the design and implementation stages of the fault model, it is necessary to verify the results of the fault model with a real-world experiment. Laboratory radiation experiments for a system, as well as radiation flight projects are currently being studied, to produce data for comparison with the estimated fault rates of the radiation fault model. This would afford us the ability to fine-tune the radiation fault model methodology and pint point any areas, which need to be added or improved. The laboratory experiments and flight experiments will also allow us to "dial-in" the estimates made for the inaccessible circuits in the system. The verification and fine-tuning of the fault model will give higher confidence in extrapolation of fault rates for next generation of processors.

Verifying Radiation Test Data

Due to its complexity, the radiation experiments performed on the PPC750 require verification. One way top approach this is to perform multiple types of experiments and compare the end results (SEU rates per bit-type). Currently,

experimental procedures are being carried out by the JPL radiation testing. There are radiation experiments being carried out in conjunction with IROC/TIMA, University of Illinois, as well as Center for Reliable Computing at Stanford University.
There is a level of uncertainty in the model used to convert radiation experiment data to space environment SEU rates. These results will be verified by the flight experiments currently being planned.

Verifying Translation of Thresholds to Environments

In addition to verification of the model and the radiation data, it is necessary to watch for any additional effects, which might have been overlooked. One of these effects, which we are currently in the process of incorporating in the fault model, is the frequency and effects of SEFIs. A flight experiment will provide much needed verification of this wide category of SEU event.

5. RESULTS

Figure 12 is an example of the output chart produced by the REE Radiation Fault Model for a number of possible NASA mission environments for specific to the FGT. It is important to realize that these fault rates are yet to be validated in a flight experiment, however they do provide an insight to the possible use of COTS based systems in relatively low-radiation environments, such as the surface of Mars and Low Earth Orbits. The rates were calculated using SEU rates such as the ones depicted in Figure 7.

Average Rates (Faults/Hour)										
Environment	Interplanetary Space	Interplanetary Space	Interplanetary Space	Interplanetary Space	600km-98°	600km-98°	600km-98°	600km-98°	Mars Surface	600km-28°
Solar Min/Max	Solar Minimum	Solar Minimum	Solar Minimum	Solar Minimum	Solar Min	Solar Max	Solar Min	Solar Max	N/A	N/A
Flare Status	No Flare	No Flare	Design Case Flare	Design Case Flare	No Flare	No Flare	Solar Flare	Solar Flare	N/A	N/A
Shielding	100 Mil (Al)	100 Mil (Al)	100 Mil (Al)	100 Mil (Al)	100 Mil (Al)	100 Mil (Al)	100 Mil (Al)	100 Mil (Al)	100 Mil (Al)	100 Mil (Al)
Node CPU (w/o Caches)	0.04	0.01	28.46	28.43	0.05	0.02	5.89	31.27	0.00	0.03
L1 Cache	0.05	0.01	37.40	37.37	0.06	0.02	7.88	5.88	0.00	0.04
RAM per node CPU	0.01	0.00	7.94	7.93	0.01	0.00	1.67	7.87	0.00	0.01
L2 Cache	0.00	0.00	1.03	1.03	0.00	0.00	0.22	1.67	0.00	0.00
Node Controller CPU	0.03	0.01	22.10	22.08	0.04	0.01	4.61	0.22	0.00	0.02
Node Controller Ram	0.01	0.00	4.20	4.20	0.01	0.00	0.88	4.60	0.00	0.00
Network Interface Units per node(2)	0.01	0.00	6.87	6.86	0.01	0.01	1.14	0.88	0.00	0.01
Network Switch Per Node(1)	0.01	0.00	3.17	3.17	0.00	0.00	0.53	1.14	0.00	0.01
PCI Bus Controller	0.00	0.00	2.58	2.58	0.00	0.00	0.43	0.52	0.00	0.00
Node Controller FPGA	0.00	0.00	1.18	1.18	0.00	0.00	0.24	0.43	0.00	0.00
watchdog,clock,EEP ROM,PHRC	0.00	0.00	2.63	2.63	0.00	0.00	0.44	0.24	0.00	0.00
Single CPU	0.10	0.03	74.83	74.76	0.13	0.05	15.66	15.64	0.01	0.08
Per Node	0.25	0.07	192.40	192.21	0.31	0.12	39.57	39.53	0.02	0.21
Per System (20 nodes)	4.97	1.33	3852.98	3849.34	6.27	2.39	792.27	791.37	0.34	4.14

Figure 12 - Fault Rates For System, And Subsystem Components For Various Space Environments And Solar Conditions

6. CONCLUSION AND FUTURE WORK

We have a usable model for a state-of-the-art processor (PPC750). However, it is important to realize that the REE Radiation Fault Model is a representation of our current understanding of the composing components of the FGT. Assumptions were made, to complete a first cut model. Future validation experiments will show the validity of many of these assumptions. The REE Radiation Fault Model should be usable for a variety of environments in which future missions will operate. It can be used as a basis for predicting operational reliability and performance, as well as feasibility of COTS based systems in space. The paragraphs above, detail the structure of the REE system architecture, and the REE Radiation Fault Model with a set of preliminary fault rate results. The REE Radiation Fault Model is a generic model that should be applicable to all processors of a given family to provide preliminary fault data for a number of configurations. These fault rates can then be used to direct architecture and component selections by error propagation analysis, and fault injection experiments. It can also serve as a starting point in the development of SIFT software, for specific mission requirements . The following are a number of suggestions for future work.

- Flight experiments to validate radiation SEU data
- Flight experiments to validate REE Radiation Fault Model Rates.
- Complete characterization of the next generation of the Power PC family processor, the G4 is in progress.
- Development of the G4 radiation fault model is in progress.
- The XPC106 bridge chip is a pivotal component of the REE architecture, and SEU characterization work is in progress.
- The current REE FGT utilizes Field Programmable Gate Arrays (FPGA), which will need to be characterized.
- Work to improve the modeling methods for peripheral components is currently in progress.
- Development of Algorithm Based Fault Tolerance Fault Tolerance (ABFT) software, as well as SIFT software are in progress.
- SEFI data is preliminary, and work to improve and characterize SEFIs is in progress.

7. ACKNOWLEDGEMENTS

This work was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under contract to the National Aeronautics and Space Administration. Funding was provided by NASA's High Powered Computing and Communication program, via the REE project.

8. REFRENCES

[1] J. Beahan, A. Johnston, L. Edmonds, D Katz, R. Ferraro, R. Some, "Detailed Radiation Fault Modeling of the Remote Exploration and Experimentation (REE) First Generation Testbed Architecture " IEEE Aero, 1999

[2] G. Swift, S. Guertin, F. Farmanesh, F. Irom, D. Millward, "Single-Event Upset in the Power PC750 Microprocessor" IEEE TNS Dec. 2001

[3] G. Swift, S. Guertin, F. Farmanesh, Jet Propulsion Laboratory, "Measurements of the PowerPC750 Upset Susceptibility to Protons and Heavy Ions" RADECS 2000

[4] J.L.Barth, Modeling Space Radiation Environments, Section 1 of the Short Course presented at the IEEE Nuclear and Space Radiation Effects Conference, Snowmass, Colorado, July 21, 1997

[5] G. Swift, S. Guertin Jet Propulsion Laboratory, "In-Flight Observations of Multiple-Bit Upset in DRAMs" Nuclear and Space Radiation Effects Conference, 2000

[6] JPL Electronic Parts Engineering RadData Server http://radnet.jpl.nasa.gov/

[7] A. H. Johnston, Jet Propulsion Laboratory, "Radiation Effects in Advanced Microelectronic Technologies," IEEE Trans. Nucl. Sci., 45, p. 1339-1354, 1998.

Arbi Karapetian *is a System Engineer for the Remote Exploration and Experimentation Project at the Jet Propulsion Laboratory. His background is in low temperature solid-state physics, electrometrical engineering, and spacecraft system architecture. He holds a B.S. in Solid State Physics from the University of California at Los Angeles.*

Raphael Some *is the Chief Engineer for the Remote Exploration and Experimentation Project at the Jet Propulsion Laboratory. Previously, at JPL, his assignments included management of the Smart Sensors, Sensor Web and the X2000 Avionics Future-Deliveries tasks. His experience prior to JPL includes the development of fault tolerant space based supercomputers as well as a variety of avionics and signal processing systems for both commercial and military applications. He holds a BSEE from Rutgers University*

John Beahan *is System Engineer for the Remote Exploration and Exploration Project at the Jet Propulsion*

Laboratory. His research interests include real-time systems, parallel and distributed systems, fault tolerance, robotics, nontraditional programming languages and genetic algorithms. He has served as system engineer and software lead on projects including an advanced computer packaging space flight experiment, two telerobot systems for satellite servicing, and a distributed fault-tolerant middleware layer for spacecraft applications. He holds a B.S. in Engineering and Applied Science from the California Institute of Technology.